



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/750,144	12/29/2000	Steve Lewontin	042933/301045	6775
826 7590 12/29/2008 ALSTON & BIRD LLP BANK OF AMERICA PLAZA 101 SOUTH TRYON STREET, SUITE 4000 CHARLOTTE, NC 28280-4000			EXAMINER DESAI, RACHNA SINGH	
			ART UNIT 2176	PAPER NUMBER
			MAIL DATE 12/29/2008	DELIVERY MODE PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary

Application No.

09/750,144

Applicant(s)

LEWONTIN, STEVE

Examiner

RACHNA S. DESAI

Art Unit

2176

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 09 September 2008.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-29 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-29 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☐ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
- 4) ☐ Interview Summary (PTO-413)
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: _____

DETAILED ACTION

1. This action is responsive to communications: Amendments and Arguments filed on 09/09/08.
2. Claims 1-29 are pending. Claims 22-29 are newly added claims.

Claim Rejections - 35 USC § 101

3. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

4. Claims 26-29 are rejected under 35 U.S.C. 101 because they are considered software per se. Computer programs may be explicitly claimed as, for example, a series of code or instructions for performing functions or may be implicitly claimed as, for example, a system, a module or an apparatus. Where there is no evidence in the specification that a means which may be interpreted as software, hardware or combinations thereof necessarily includes hardware, it will be interpreted in its broadest reasonable sense as a software means, which is the case here.

Thus a claim to functional descriptive material, including computer programs, per se, is not patent eligible subject matter

Claim Rejections - 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

6. **Claims 1-3, 7, 10-13, 17, 20-22, 25-26 and 29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Polonsky et al., US 7,072,984 B1, 07/04/06 (filed 04/25/01, provisional filed 04/26/00), in view of Amano et al. ("Amano"), U.S. Patent No. 6,003,033.**

Regarding independent claim 1, Polonsky discloses a method of representing a document written in a markup language and on a mobile terminal/apparatus adapted to receive said document and render said document on said display (see Abstract), the method comprising:

➤ *providing a virtual node tree describing the structure of the data types, with each one of the nodes in the virtual node tree respectively corresponding to one element of a specific data type in the document* (see Fig. 2 and column 9, lines 3-67 and column 10, lines 1-22, column 14, lines 17-26, column 11, lines 47-67 and column 16, lines 1-22: Polonsky teaches a DOM tree. DOM converts the document in memory into a hierarchical node tree that looks like a database

record, wherein each node corresponds to an element in the document. DOM provides an interface to access and change the content and structure of an XML file);

Provisional Citation: figure 6, pages 12-15, and page 17-18.

➤ for each one of the nodes in the virtual node tree, providing a data array including information identifying the relationship of the node to other nodes in the virtual node tree and a reference indicating the location of data corresponding to the node (see column 2, lines 54-65 and column 16, lines 23-30): Polonsky teaches a data array including information identifying the relationships and locations of nodes. Moreover, it was commonly known to those of ordinary skill in the art and would have been obvious at the time the invention was made to a person having ordinary skill in the art that DOM trees can be implemented as an array (or as a linked list) for the motivational purpose of implementing a organized data structure. Popular browsers such as Internet Explorer and Mozilla used the array implementation for their DOM trees); and

Provisional Citation: The Provisional discloses a DOM that identifies each node in the document using a unique value (see Provisional on page 9) and assigns the unique weight/priority value based on the node's relationships and location relative to other objects (see pages 12-14 where the process is implemented using a stack array and a node table array). See also pages 16-18. The DOM tree array therefore contains data information describing the tree's structure, the tree's dependencies (root, parent, child, etc) and references to information content data (font attribute importance) with unique weight and priority values. Moreover, it was commonly known to those of ordinary skill in the art at the time of the invention was made that DOM trees were implemented using an array (or a linked list) for the purpose of implementing an organized data structure of markup documents. Internet browsers such as Internet Explorer and Mozilla specifically used the array implementation for their DOM trees. Therefore, the Provisional discloses, or at the very least made obvious to one of ordinary skill in the art, a data

array including information identifying the relationship of a node to other nodes in the virtual tree and a reference indicating the location of data corresponding to the node.

➤ *obtaining, by a set of software components in the apparatus, the data corresponding to the nodes using the reference included in the data array (see column 2, lines 54-64 and columns 15-16, lines 66-67 and 1-9 respectively):*

Based on the nodes of the object tree, the Polonsky method generates an array of primitive data types for efficiently developing an optimized standard structure).

Provisional Citation: See page 18.

Polonsky does not explicitly disclose *providing a virtual node tree describing the structure of the data types in the document but not containing actual document data.*

However, Amano discloses:

providing a virtual node tree describing the structure of the data types in the document but not containing actual document data (see Fig. 5; col. 5 lines 50-59 et seq. → i.e. Tree Skeleton), and additionally, also discloses:

for each one of the nodes in the virtual node tree, providing a data array including information identifying the relationship of the node to other nodes in the virtual node tree and a reference indicating the location of data corresponding to the node (see col. 10 lines 29-40 et seq.; Fig. 18).

Since Polonsky and Amano (hereinafter "Polonsky-Amano") are both from the same field of endeavor, the motivational purpose and advantage is the simplification in defining a tree and generating a data structure corresponding to the tree in memory with the resultant description as disclosed by Amano (see col. 13 lines 1-6) would have been recognized in the pertinent art of Polonsky. It would have been obvious at the time the invention was made to a person having ordinary skill in the art to modify the teaching of Polonsky with the teachings of Amano to include a virtual node tree, not containing actual document data.

Independent claim 10 incorporates substantially similar subject matter as independent claim 1 and is rejected along the same rationale.

Regarding claims 2 and 13, Polonsky discloses wherein the data in the document is stored in a document block in memory (see column 15, lines 47-52 and column 28, lines 37-49).

Regarding claims 3, 11, and 12, Polonsky discloses wherein the document is written in XML or a variation of XML and displayed on an XML Browser (see column 2, lines 32-38 and column 4, lines 46-59).

Regarding claims 7 and 17, Polonsky-Amano further discloses an array indicating whether a node is a sibling or child (see columns 15-16, lines 66-67 and 1-9

Art Unit: 2176

respectively and column 26, lines 42-52), but does not specifically teach indicating whether or not the node is the last sibling in a list of siblings and does not specifically teach a child index and a sibling index in the data array.

However, it was commonly known to those of ordinary skill in the art and would have been obvious at the time the invention was made to a person having ordinary skill in the art that the position of a node can be calculated by its arrangement in the data array relative to its siblings for the purpose of indicating whether or not the node is the last sibling in a list of siblings. It was also commonly known to those of ordinary skill in the art and would have been obvious at the time the invention was made to a person having ordinary skill in the art that a child and sibling index can be created using data arrays to further clarify the hierarchical standing of particular nodes.

Regarding claims 20 and 21, Polonsky-Amano discloses storing data arrays in the memory of the mobile phone/terminal (see column 2, lines 54-64 and column 16, lines 10-15).

Regarding claims 22 and 25, claims 22 and 25 are drawn the computer readable storage medium having the cod for carrying out the method of claims 1 and 7 respectively. Accordingly, claims 22 and 25 are rejected under the same rationale used in claims 1 and 7 respectively above.

Regarding claims 26 and 29, claims 26 and 29 are drawn to the apparatus comprising the means for the steps of claims 1 and 7 respectively. Accordingly, claims 26 and 29 are rejected under the same rationale used in claims 1 and 7 respectively above.

7. Claims 4-6, 8, 9, 14-16, 18-19, 23-24 and 27-28 are rejected under 35 U.S.C. 103(a) as being unpatentable over Polonsky et al., US 7,072,984 B1, 07/04/06 (filed 04/25/01, provisional filed 04/26/00), in view of Amano et al. ("Amano"), U.S. Patent No. 6,003,033 as applied to claims 1-3, 7, 10-13, 17, 20, and 21 above, in further view of, Call, U.S. Patent Application Publication No. 2002/0143521.

Regarding claims 4 and 14, Polonsky-Amano discloses the method with respect to independent claims 1 and 10 above, but does not specifically disclose data arrays including a flags field.

However, Call discloses the use of flags to uniquely identify a selected rule in a manner tailored to the needs of the portion of the XML document (see Call [0362]) for the purpose of signaling a particular condition or status.

It would have been obvious at the time the invention was made to a person having ordinary skill in the art to modify the teaching of Polonsky-Amano with the teachings of Call to include the use of flag fields to uniquely identify a selected rule in a

manner tailored to the needs of the portion of the XML document (see Call [0362]) for the purpose of signaling a particular condition or status.

Regarding claims 5 and 15, Polonsky-Amano further discloses an array indicating whether a node is a sibling or child (see columns 15-16, lines 66-67 and 1-9 respectively and column 26, lines 42-52), but does not specifically teach indicating whether or not the node is the last sibling in a list of siblings and does not specifically teach a child index and a sibling index in the data array.

However, it was commonly known to those of ordinary skill in the art and would have been obvious at the time the invention was made to a person having ordinary skill in the art that the position of a node can be calculated by its arrangement in the data array relative to its siblings for the purpose of indicating whether or not the node is the last sibling in a list of siblings. It was also commonly known to those of ordinary skill in the art and would have been obvious at the time the invention was made to a person having ordinary skill in the art that a child and sibling index can be created using data arrays to further clarify the hierarchical standing of particular nodes.

Regarding claims 6 and 16, Polonsky-Amano does not specifically disclose the method wherein a flag in the flags field identifies the type of the node data. However, Call discloses that the header information for each node identifies the data type of the node (see Call [0368]; see also Figure 6).

It would have been obvious at the time the invention was made to a person having ordinary skill in the art to modify the teaching of Polonsky-Amano with the teachings of Call to include the use of flag fields to uniquely identify the data type of the node for the purpose of signaling a particular condition or status of the data.

Regarding claims 8 and 18, Polonsky-Amano discloses the method and mobile phone with respect to independent claims 1 and 10 as discussed above, but does not specifically teach whether the data arrays have a fixed length.

However, it was commonly known to those of ordinary skill in the art and would have been obvious at the time the invention was made to a person having ordinary skill in the art that data arrays can be fixed for the purpose of holding a preset. Furthermore, Call discloses fixed length data as an addressable array to provide efficient data manipulation functions typically performed by hierarchical object oriented data systems, including systems conforming to the Document Object Model widely used for storing and manipulating XML and HTML character data (see Call [0016] and [0017]) for the purpose of compact data representation to preserve storage space (see Call [0013]).

It would have been obvious at the time the invention was made to a person having ordinary skill in the art to modify the teaching of Polonsky-Amano with the teachings of Call to include the representation of a document structure written in a markup language, wherein the data arrays have a fixed length for the purpose of compact data representation to preserve storage space.

Regarding claims 9 and 19, Polonsky-Amano discloses the method and mobile phone with respect to independent claims 1 and 10 as discussed above, but does not specifically teach whether the data arrays have a variable length.

However, it was commonly known to those of ordinary skill in the art and would have been obvious at the time the invention was made to a person having ordinary skill in the art that data arrays can be variable for the purpose of expanding number of objects. Furthermore, Call discloses variable length data as an addressable array to provide efficient data manipulation functions typically performed by hierarchical object oriented data systems, including systems conforming to the Document Object Model widely used for storing and manipulating XML and HTML character data (see Call [0016] and [0017]) for the purpose of compact data representation to preserve storage space (see Call [0013]).

It would have been obvious at the time the invention was made to a person having ordinary skill in the art to modify the teaching of Polonsky-Amano with the teachings of Call to include the representation of a document structure written in a markup language, wherein the data arrays have a variable length for the purpose of compact data representation to preserve storage space.

Regarding claims 23-24, claims 23-24 are drawn the computer readable storage medium having the cod for carrying out the method of claims 5-6 respectively. Accordingly, claims 23-24 are rejected under the same rationale used in claims 5-6 respectively above.

Regarding claims 27-28, claims 26 and 29 are drawn to the apparatus comprising the means for the steps of claims 5-6 respectively. Accordingly, claims 27-28 are rejected under the same rationale used in claims 5-6 respectively above.

Response to Arguments

8. Applicant's arguments and remarks have been fully considered but are not persuasive. Please see the response to each argument in the sections below.

A. Simon Hunt and Polonsky.

Beginning on page 7 and continuing through page 8 of the Response, Applicant provides an overview of the prosecution history of the application. Applicant concludes by requesting that for each citation of the Polonsky patent, the corresponding citations to the Polonsky provisional be provided. It is noted, that these portions were provided in the response to arguments portion of the Non-final Rejection dated 08/12/08; however, those portions are reproduced again in the interest of furthering prosecution. It is further noted, that only those portions that are challenged by the Applicant in their Request for Reconsideration as not being taught by the Provisional are recited in the rejections above and relevant portions below.

B. The Claimed Invention is Patentable

1. Claims 1-3, 7, 10-13, 17, 20, and 21 are patentable over Polonsky in view of Amano.

a. Polonsky Cited Disclosure is not Prior Art to the Claimed Invention.

Applicant reiterates several of the previous arguments with respect to the Simon Hunt patent to the Polonsky patent.

On pages 9-13, Applicant argues the feature of the QDOM or its technique for generating a representation of a DOM tree is not disclosed in the provisional application. Specifically, Applicant states the provisional **does** teach normalizing DOM tree, but does not realize a data array including information identifying the relationship of a node to other nodes in the virtual node tree and a reference indicating the location of data corresponding to the node.

Examiner disagrees.

By Applicant's own admission, the provisional teaches a DOM tree node. See page 2, third paragraph; pg. 3, first paragraph; and pg 8 of the provisional application where a method and apparatus for Internet communication for mobile appliances wherein a markup language (XML, HTML, SGML, etc) is organized into a DOM tree. DOM is a specification for a programming interface (API) from the W3C that allows programs and scripts to update the content, structure, and style of HTML and XML documents by creating a hierarchical virtual tree of the document that looks like a database record. The Provisional discloses a DOM that identifies each node in the

document using a unique value (see Provisional on page 9) and assigns the unique weight/priority value based on the node's relationships and location relative to other objects (see pages 12-14 where the process is implemented using a stack array and a node table array). The DOM tree array therefore contains data information describing the tree's structure, the tree's dependencies (root, parent, child, etc) and references to information content data (font attribute importance) with unique weight and priority values.

Moreover, it was commonly known to those of ordinary skill in the art at the time of the invention was made that DOM trees were implemented using an array (or a linked list) for the purpose of implementing an organized data structure of markup documents. Internet browsers such as Internet Explorer and Mozilla specifically used the array implementation for their DOM trees.

Therefore, the Provisional discloses, or at the very least made obvious to one of ordinary skill in the art, a data array including information identifying the relationship of a node to other nodes in the virtual tree and a reference indicating the location of data corresponding to the node.

Applicant continues to challenge the teachings in the provisional application but makes several admissions worth noting. Applicant admits the provisional application **does** teach a DOM tree to a normalization process (see page 10). Applicant admits the provisional **does disclose** translating content that is organized into a DOM tree by a DOM Layer then subjected to a Normalization Layer (see page 11). Applicant further admits on page 12, second full paragraph that **one could argue the DOM layer of the**

provisional corresponds to the QDOM. Applicant also admits on page 11, last paragraph that the provisional does support the normalizer.

Applicant argues that even given the above teachings of the provisional application, the provisional does not describe the DOM Layer in a manner that supports the functions of the QDOM because the QDOM extends the W3C DOM interface definition to an efficient model that provides high speed parsing, storage, and access by minimizing memory resource requirements. Further, Applicant argues that the normalizer does not realize a data array including information identifying the relationship of a node to other nodes in a virtual tree node and a reference indicating the location of data corresponding to the node.

Examiner disagrees.

DOM is a specification for a programming interface (API) from the W3C that allows programs and scripts to update the content, structure, and style of HTML and XML documents by **creating a hierarchical virtual tree** of the document that looks like a database record. The Provisional discloses a DOM that identifies each node in the document using a unique value (see Provisional on page 9) and assigns the unique weight/priority value based on the **node's relationships and location relative to other objects** (see pages 12-14 where the process is implemented using a stack array and a node table array). The **DOM tree array therefore contains data information describing the tree's structure, the tree's dependencies (root, parent, child, etc) and references to information content data (font attribute importance) with unique weight and priority values.**

Moreover, it was commonly known to those of ordinary skill in the art at the time of the invention was made that DOM trees were implemented using an array (or a linked list) for the purpose of implementing an organized data structure of markup documents. Internet browsers such as Internet Explorer and Mozilla specifically used the array implementation for their DOM trees.

Therefore, the Provisional discloses, or at the very least made obvious to one of ordinary skill in the art, a data array including information identifying the relationship of a node to other nodes in the virtual tree and a reference indicating the location of data corresponding to the node.

b. The Polonsky Patent does not teach/suggest elements of the claimed invention.

Applicant argues Polonsky does not teach for each node in a virtual tree, a data array including information identifying the relationship of the node to other nodes in the virtual tree, and a reference indicating the location of data corresponding to that node. Applicant argues DOM trees could not be implemented as an array and the office has taken official notice of this fact.

Examiner disagrees that any official notice has been taken as Polonsky expressly states DOM trees can be implemented as a data array including information identifying relationships and locations of nodes in column 2, lines 54-65 where he states "based on the nodes of the object tree, the QDOM generates an **array** of primitive data types for efficiently developing an optimized standard structure for use by a normalizer

or other processing modules". Again in column 16, lines 23-30, Polonsky states, "the **QDOM array** references a content buffer when information content that is not well known is used. **The QDOM array** preferably stores the start and end positions of text in the content buffer. . ." Therefore, no official notice has been taken as the QDOM array is expressly taught by the Polonsky reference.

Applicant argues while the Amano patent teaches generating a data structure whereby parent nodes are connected to child nodes, the Amano patent does not disclose the data structure includes a reference indicating the location of data corresponding to the node.

Examiner disagrees because Amano teaches the **pointers reference the location of each of the nodes in the node tree** (see Abstract).

Therefore, both Amano and Polonsky teach the data structure includes a reference indicating the location of data corresponding to the node.

c. *The Amano Patent does not teach elements of the claimed invention.*

Applicant argues Amano does not teach *for each node of the virtual tree node tree, a reference including the location of data corresponding to the node*. Specifically, Applicant states that while Amano teaches generating a data structure where parent nodes are connect to child notes, Amano does not teach the data structure includes, for each node in the virtual node tree, a reference indicating the location of data corresponding to the node.

Examiner disagrees.

Applicant states, ***“one could argue that the Amano patent discloses, for a parent node, a reference indicating the location of data corresponding to the child node”***; however, Applicant believes Amano does not teach the data structure for the parent node includes a reference indicating the location of data corresponding to the respective parent node. Examiner disagrees because Amano teaches the **pointers reference the location of each of the nodes in the node tree** (see Abstract).

3. No Motivation to Combine Polonsky and Amano

On pages 17-19, Applicant argues one skilled in the art would not be motivated to combine the teachings because there is no desirability of the combination.

Examiner disagrees.

Polonsky and Amano are not only from the same field of endeavor (data structure corresponding to a node tree in computer memory) but the advantage of defining a tree and generating a data structure corresponding to the tree in memory with the description disclosed by Amano (see col. 13, lines 1-6) within Polonsky would be to simplify and present tree data structure information for electronic devices.

On pages 19-20, Applicant argues:

2. Claims 4, 6, 8, 9, 14, 16, 18, and 19 are patentably distinct from Polonsky in view of Amano, and further in view of Call.

Applicant submits the claims are patentable by virtue of their dependency on claims 1 and 10. The reasons provided above for using the Polonsky reference with respect to claims 1 and 10 also apply to the dependent claims 4, 6, 8, 9, 14, 16, and 18-19.

In view of the comments above, the rejections are maintained.

Conclusion

9. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the date of this final action.

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to RACHNA S. DESAI whose telephone number is (571)272-4099. The examiner can normally be reached on M-F (8:30AM-6:00PM).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Doug Hutton can be reached on 571-272-4137. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Rachna S Desai/
Primary Examiner, Art Unit 2176
12/23/08